Klasifikasi Kualitas Air Sungai menggunakan Random Forest, SVC Dan Logistic Regression

[1] Sheila Tania [2]Eriko Levino Sasmitra [3]Dave Keane Wijaya [4] Nelson

[1] [2] [3] [4]Teknik Informatika Universitas Tarumanagara
Jl. Letjen S. Parman No. 1, Tomang, Grogol petamburan, RT.6/RW.16, Tomang, Grogol petamburan, Kota Jakarta Barat, Daerah Khusus Ibukota Jakarta 11440, Indonesia
email : [1] sheila.535220028@stu.untar.ac.id, [2] eriko.535220014@stu.untar.ac.id, [3] dave.535220022@stu.untar.ac.id, [4] nelson.535220021@stu.untar.ac.id

ABSTRAK

Studi ini bertujuan untuk mengklasifikasikan kualitas air sungai menggunakan tiga algoritma machine learning: Random Forest, Support Vector Classifier (SVC), dan Logistic Regression. Tujuan dari penelitian ini adalah untuk membandingkan kinerja ketiga algoritma tersebut berdasarkan accuracy, precision, recall, dan F1-score guna menentukan model yang paling sesuai untuk pemantauan kualitas air. Dataset yang digunakan diperoleh dari portal Satu Data Jakarta, yang berisi pengukuran kualitas air dari 23 sungai berdasarkan parameter fisik dan kimia seperti pH, dissolved oxygen (DO), biochemical oxygen demand (BOD), total dissolved solids (TDS), suhu, dan kekeruhan. Data tersebut melalui proses preprocessing melalui pembersihan, categorical encoding, dan feature scaling sebelum digunakan untuk pelatihan dan pengujian model. Hasil penelitian menunjukkan bahwa algoritma Random Forest mencapai accuracy tertinggi (93,33%) dan F1-score sebesar 0,87, diikuti oleh SVC dan Logistic Regression. Logistic Regression menghasilkan precision tertinggi, sementara Random Forest memberikan keseimbangan terbaik antara precision dan recall. Temuan ini menunjukkan bahwa Random Forest merupakan algoritma yang paling efektif dan adaptif untuk klasifikasi kualitas air sungai, menawarkan kinerja yang andal untuk mendukung pemantauan lingkungan dan pengelolaan air yang berkelanjutan.

Key words

Logistic Regression, Machine Learning, Random Forest, Support Vector Classifier, Quality Classification

1. Pendahuluan

1.1 Latar Belakang Masalah

Air adalah sumber daya alam yang sangat penting bagi kehidupan manusia, hewan, dan tumbuhan. Selain digunakan untuk keperluan sehari-hari seperti minum, memasak, dan mencuci, air juga sangat penting dalam kegiatan ekonomi seperti pertanian, industri, dan pembangkit listrik. Namun, dengan bertambahnya aktivitas manusia di berbagai bidang seperti industri, perkotaan, dan pertanian yang intensif, kualitas air di berbagai sungai mengalami penurunan yang sangat signifikan. Limbah dari industri, sisa pertanian seperti pupuk dan pestisida, serta pembuangan sampah rumah tangga yang tidak diolah secara benar, menjadi penyebab utama terjadinya pencemaran air.

Pencemaran air tidak hanya merusak ekosistem perairan, tetapi juga membahayakan kesehatan manusia. Air yang tercemar dapat menyebabkan penyakit seperti diare, keracunan logam berat, dan gangguan pada kulit. Selain itu, penurunan kualitas air juga mempengaruhi ketersediaan air bersih dan keberlanjutan sumber daya air di masa depan. Karena itu, penting untuk memantau, mengevaluasi, dan mengklasifikasikan kualitas air sungai secara akurat agar lingkungan tetap seimbang dan pengelolaan air bisa berkelanjutan [1], [2], [3], [4].

Belakangan ini, perkembangan teknologi machine learning memberikan solusi yang efisien dan adaptif dalam menganalisis data lingkungan. Teknologi ini mampu memproses data dalam jumlah besar dan menemukan pola dari data historis yang bisa membantu pengambilan keputusan. Dalam konteks pemantauan kualitas air, machine learning bisa digunakan untuk mengklasifikasikan dan memprediksi kondisi air berdasarkan parameter seperti pH, kadar oksigen terlarut (DO), kebutuhan oksigen biokimia (BOD), jumlah zat terlarut total (TDS), suhu, serta tingkat kekeruhan.[5], [6]. Oleh karena itu, penerapan algoritma seperti *Random Forest*, *Support Vector Machine* (SVC), dan *Logistic Regression* menjadi langkah strategis dalam mendukung pengawasan kualitas air yang berkelanjutan serta menjaga lingkungan di masa kini [7], [8], [9], [10].

1.2 Penelitian Terkait yang Sudah Ada

Beragam studi telah menggunakan pendekatan machine learning untuk memodelkan kualitas air. Berikut adalah beberapa penelitian terkait yang sudah ada;

1. Klasifikasi Kualitas Air Sungai Daerah Istimewa Yogyakarta (DIY) Menggunakan Algoritma Random Forest (2025). Penelitian ini mengembangkan model klasifikasi kualitas air sungai di Daerah Istimewa Yogyakarta menggunakan algoritma Random Forest. Metode cross-validation diterapkan dan menghasilkan akurasi rata-rata sebesar 100 % pada data pelatihan dan 91,43 % pada data pengujian. Meskipun terdapat indikasi overfitting, model terpilih (indeks ke-4) berhasil memberikan akurasi 97,93 % pada data baru, menunjukkan bahwa model tersebut mampu mengklasifikasikan kualitas air sungai di DIY dengan cukup baik.
2. *Simple Prediction of an Ecosystem-Specific Water Quality Index and the Water Quality Classification of a Highly Polluted River through Supervised Machine Learning (2022).* Penelitian ini memanfaatkan metode supervised machine learning untuk memprediksi indeks kualitas air (WQI) khusus ekosistem pada Sungai Santiago-Guadalajara. Penulis menggunakan teknik reduksi parameter (best subset selection) untuk mengurangi jumlah parameter dari 17 menjadi 12 tanpa kehilangan akurasi signifikan. Hasil regresi multiple linear regression menunjukkan RSE = 3,255 (17 parameter) dan RSE = 3,262 (12 parameter) dengan R^2 di kisaran ~0,82. Untuk klasifikasi, logistic regression dengan 17 parameter menghasilkan ~93 % klasifikasi benar, sementara versi 12 parameter mencapai ~86 %. Model generalized additive bahkan mencapai adjusted R^2 = 0,9992 dan menunjukkan presisi tinggi dalam prediksi non-linear [11], [12], [13], [14].
3. Pemodelan Mutu Kualitas Air Sungai DIY Tahun 2020 dengan Regresi Logistik Ordinal (2023). Penelitian ini bertujuan untuk mengidentifikasi faktor-faktor kimia yang berpengaruh signifikan terhadap mutu kualitas air sungai di Daerah Istimewa Yogyakarta. Data yang digunakan berasal dari hasil pengukuran Dinas Lingkungan Hidup DIY tahun 2020, meliputi parameter Total Suspended Solid (TSS), BOD, COD, dan Total Coliform. Metode yang digunakan adalah regresi logistik ordinal untuk menentukan hubungan antara tingkat pencemaran (memenuhi, tercemar ringan, sedang, dan berat) dengan variabel-variabel kimia tersebut.
4. Analisis Kualitas Air dan Limbah Pertambangan Nikel di Sungai Pesouha, Sulawesi Tenggara (2025). Penelitian ini bertujuan untuk menilai kualitas air Sungai Pesouha yang berada di kawasan pertambangan nikel Pomalaa, Kabupaten Kolaka, serta mengidentifikasi parameter utama penyumbang pencemaran. Pengujian dilakukan dengan menggunakan metode Indeks Pencemaran (IP), Indeks Kualitas Air Modifikasi Indonesia (IKA-INA), serta analisis korelasi dan regresi. Hasil penelitian menunjukkan bahwa nilai IP Sungai Pesouha berkisar antara 0,617–2,47, yang mengindikasikan kualitas air dari baik hingga tercemar ringan, sedangkan nilai IKA-INA berada pada kisaran 79,2–90,93 dengan kategori cukup baik hingga sangat baik. Parameter yang paling berpengaruh terhadap pencemaran adalah kromium heksavalen (Cr6+), besi (Fe), dan total suspended solid (TSS). Nilai korelasi antara Cr6+ dan Fe mencapai 0,91, sedangkan antara Cr6+ dan IP sebesar 0,72, menunjukkan hubungan positif yang kuat.
5. Developing river water quality prediction model incorporating reliable indexing approach (2025).Penelitian ini berfokus pada pengembangan model prediksi kualitas air sungai menggunakan algoritma machine learning dengan optimisasi hyperparameter untuk meningkatkan akurasi. Data kualitas air dikumpulkan berdasarkan parameter fisik dan kimia. Hasil penelitian menunjukkan bahwa penerapan optimisasi hyperparameter mampu meningkatkan performa model secara signifikan dibandingkan model standar, menghasilkan prediksi kualitas air yang lebih akurat dan stabil.

1.3 Perbedaan Penelitian Ini dengan Penelitian Sebelumnya

Berbagai penelitian terdahulu umumnya berfokus pada penerapan satu algoritma tertentu dalam memprediksi atau mengklasifikasikan kualitas air sungai, seperti Random Forest, Logistic Regression, atau metode indeks pencemaran. Selain itu, sebagian penelitian lebih menekankan pada analisis parameter kimia atau optimisasi model tunggal tanpa melakukan perbandingan menyeluruh tiap algoritma. Berbeda dengan penelitian-penelitian sebelumnya, penelitian ini menggunakan pendekatan yang mengkombinasikan tiga algoritma berbeda, yaitu *Random Forest*, *Support Vector Machine* (SVC), dan *Logistic Regression*, untuk menganalisis dan mengklasifikasikan kualitas air sungai.

Penelitian ini juga berfokus pada penerapan pendekatan regresi dalam menentukan kelas mutu air berdasarkan hasil prediksi nilai parameter utama seperti pH, DO, BOD, TDS, suhu, dan kekeruhan, sehingga mampu menghasilkan model yang lebih adaptif terhadap kondisi lingkungan lokal. Dengan demikian, penelitian ini memberikan kontribusi baru dalam penerapan machine learning untuk pemantauan kualitas air, sekaligus menjadi acuan dalam memilih algoritma yang paling sesuai untuk mendukung pengawasan lingkungan yang lebih efisien dan akurat.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah membuat dan membandingkan berbagai model machine learning untuk mengklasifikasikan kualitas air sungai berdasarkan parameter fisik dan kimia. Penelitian ini juga bertujuan untuk membangun model prediksi menggunakan tiga algoritma yaitu *Random Forest*, *Support Vector Machine* (SVC), dan *Logistic Regression* dalam mengidentifikasi tingkat kualitas air dari data yang tersedia. Selain itu, penelitian ini juga ingin mengevaluasi bagaimana baiknya ketiga algoritma tersebut bekerja dengan menggunakan *Confusion Matrix & Derived Metrics* dan *ROC Curve* (*Receiver Operating Characteristic Curve*) & *AUC* (*Area Under the Curve*), sehingga bisa mengetahui algoritma mana yang paling tepat untuk mengklasifikasikan kualitas air sungai. Hasil penelitian ini diharapkan dapat membantu mengembangkan metode analisis kualitas air yang lebih akurat, efisien, serta bisa digunakan sebagai dasar dalam pengambilan keputusan mengenai pengelolaan air secara berkelanjutan [15], [16].

2. Metode Penelitian

2.1 Latar Belakang Masalah

Pertama, akan dilakukan proses pengumpulan dan pra-pemrosesan data dengan mengumpulkan dataset yang akan digunakan, setelah itu data akan dibersihkan dengan membuang data dengan nilai yang kosong, data-data duplikat, dan normalisasi data. Tahap ini penting karena kualitas dari data dapat mempengaruhi kinerja model. Setelah pengumpulan dan pra-pemrosesan, data dari dataset akan dibagi menjadi data training dan data testing

Berikutnya akan dilakukan training dan testing, disini model dilatih menggunakan data training untuk mengenali pola dan hubungan-hubungan antar variabel. Selanjutnya, model akan diuji menggunakan data testing untuk mengetahui kemampuan model dalam melakukan prediksi terhadap data-data baru.

Terakhir akan dilakukan evaluasi yang bertujuan untuk mengukur performa dari model menggunakan sejumlah metode evaluasi seperti akurasi (accuracy), presisi (precision), recall, dan F1-score. Dalam tahap ini, peneliti dapat menilai seberapa baik model bekerja serta melakukan perbandingan antara metode-metode yang digunakan dan menentukan model terbaik yang paling sesuai dengan kebutuhan penelitian.

2.2 Pengumpulan dan Pra-Pemrosesan Data

Dataset yang digunakan dalam penelitian ini diambil dari Satu Data Jakarta (https://satudata.jakarta.go.id), yang berisi data pemantauan kualitas air sungai di wilayah DKI Jakarta untuk periode tahun 2024. Dataset ini mencakup sejumlah variabel seperti latitude, longitude, jenis parameter, baku mutu, dan hasil pengukuran.

Untuk memastikan data yang digunakan memiliki variasi yang memadai, penelitian ini memanfaatkan data dari 23 sungai berbeda, yaitu: Angke, Cakung, Cengkareng, Blencong, Buaran, Cideng, Sunter, Kalibaru Timur, Kalibaru Barat, Petukangan, Ciliwung, Jati Kramat, Cipinang, Grogol, Mookervart, Pesanggrahan, Mampang, Sekretaris, Sepak, Krukut, Kanal Timur, Kamal, dan Tarum Barat.

Sebelum data digunakan dalam model, dilakukan serangkaian tahapan pra-pemrosesan meliputi:

1. Data Cleaning – Menghapus baris yang memiliki nilai kosong atau tidak valid untuk menjaga integritas dataset.
2. Categorical Encoding – Mengonversi variabel kategorikal (seperti nama sungai atau jenis parameter) menjadi bentuk numerik agar dapat diproses oleh algoritma *machine learning*.
3. Feature Scaling – Menyelaraskan skala setiap fitur agar memiliki rentang nilai yang seragam, terutama penting untuk metode seperti *Support Vector Machine (SVM)* [17].

Langkah-langkah ini dilakukan agar data menjadi lebih bersih, konsisten, dan siap digunakan pada proses pelatihan dan pengujian model.

2.3 Data *Training*

Data trainning adalah sekumpulan data yang digunakan untuk melatih model agar dapat mengenali pola dan hubungan antar variabel. Dalam konteks penelitian ini, model dilatih menggunakan 80% dari total dataset yang telah melalui proses pra-pemrosesan. Proses pra-pemrosesan tersebut meliputi pembersihan data, normalisasi, dan pemisahan atribut agar data yang digunakan benar-benar siap untuk dipelajari oleh model. Proses pelatihan dilakukan dengan tujuan agar algoritma dapat "belajar" dari contoh-contoh data yang ada.

Model akan menyesuaikan bobot dan parameter internalnya sehingga mampu menghasilkan prediksi yang mendekati nilai aktual. Selain itu, selama tahap ini dilakukan evaluasi performa model secara bertahap untuk memastikan proses pembelajaran berjalan optimal dan tidak mengalami overfitting terhadap data pelatihan.. Tahap pelatihan ini juga mencakup proses *tuning* sederhana terhadap parameter algoritma untuk mendapatkan performa terbaik dari masing-masing model [1].

2.4 Data *Testing*

Data Testing adalah sekumpulan data yang akan digunakan untuk mengevaluasi kinerja dari model-model setelah mereka dilatih menggunakan data training, agar model dapat mengenali pola-pola, hubungan, atau struktur di dalam data.[1]

Data Testing tidak pernah digunakan selama proses pelatihan, hal ini dilakukan agar evaluasi dapat mencerminkan kemampuan dari model terhadap data baru yang belum pernah dilihat sebelumnya.

Dalam penelitian ini, *data testing* yang akan digunakan akan berupa 20% dari dataset.

2.5 Metode *Logistic Regression*

Metode *Logistic Regression* adalah sebuah metode algoritma dasar untuk memodelkan hubungan antara satu atau lebih variabel independen (fitur, prediktor) dengan variabel dependen yang bersifat kategorik biner (dua kelas, misalnya 0 atau 1, "sukses/gagal", "ya/tidak").[18]

Metode *Logistic Regression* digunakan dalam penelitian ini untuk melakukan klasifikasi kualitas air sungai menjadi "baik" dan "buruk", berdasarkan data historik dari 4 kolom yaitu *latitude, longitude,* jenis parameter, dan parameter

2.6 Metode Random Forest

Random Forest adalah algoritma berbasis ensemble learning yang menggabungkan beberapa decision tree untuk menghasilkan prediksi yang lebih stabil dan akurat [19].

Setiap pohon keputusan dalam Random Forest dilatih dengan subset data dan fitur yang berbeda, sehingga variasi di antara pohon membantu mengurangi *overfitting*. Dalam penelitian ini, *Random Forest* digunakan untuk mengklasifikasikan data kualitas air berdasarkan parameter seperti pH, suhu, BOD, DO, dan TDS. Kelebihan utama metode ini adalah kemampuannya dalam menangani data *nonlinier* dan memberikan *feature importance* yang dapat menunjukkan parameter mana yang paling berpengaruh terhadap hasil klasifikasi. Selain itu, algoritma ini juga memiliki keunggulan dalam hal kecepatan pelatihan, ketahanan terhadap *noise* pada data, serta performa yang konsisten meskipun data memiliki distribusi yang kompleks. Dengan karakteristik tersebut, *Random Forest* menjadi salah satu metode yang andal dalam analisis kualitas air dan perbandingan performa model klasifikasi.

2.7 Metode *SVC*

Metode *Support Vector Machine*(SVM) adalah metode *supervised learning* yang mencari garis(untuk 2D) atau bidang (untuk 3D) yang memisahkan kelas dengan margin maksimal, yaitu jarak sejauh mungkin dari hyperplane ke titik data terdekat di setiap kelas.[20]

Dengan prinsip tersebut, SVM berupaya membentuk batas keputusan yang optimal agar mampu memisahkan data dari dua kelas dengan tingkat kesalahan seminimal mungkin.

Dalam penelitian ini, metode SVC (*Support Vector Classifier*) digunakan untuk membandingkan performa model ini dengan *Logistic Regression* dan *Random Forest* dalam mengklasifikasikan kualitas air sungai. SVM dipilih karena kemampuannya menangani data berdimensi tinggi serta keunggulannya dalam menjaga keseimbangan antara kompleksitas model dan generalisasi, sehingga diharapkan mampu memberikan hasil klasifikasi yang akurat pada data kualitas air yang memiliki karakteristik bervariasi.

2.8 Metode Evaluasi

Dalam penelitian sejumlah metode evaluasi yang digunakan, metode-metode ini digunakan untuk melihat performa dan kemampuan mengklasifikasi kualitas dari air-air sungai, metrik-metrik yang akan dipakai meliputi Confusion Matrix & Derived Metrics dan ROC & AUC.

Confusion Matrix merupakan metode evaluasi yang digunakann untuk menilai kinerja model klasifikasi dengan membandingkan hasil prediksi model terhadap label aktual. Matriks ini terdiri dari empat komponen utama, yaitu:

1. *True Positve* (TP): data yang benar terindentifikasi dengan kelas positif.
2. *True Negative* (TN): data yang benar teridentifikasi sebagai kelas negatif.
3. *False Positive* (FP): data negatif yang keliru diprediksi sebagai positif.
4. *False Negative* (FN): data positif yang keliru di prediksi sebagai negatif.

Berdasarkan keempat komponen tersebut, terdapat beberapa ukuran yang digunakan untuk menilai performa model, diantaranya:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Accuracy bertujuan untuk mengukur seberapa banyak prediksi model yang benar dibandingkan dengan keseluruhan jumlah data, Accuracy akan menunjukkan proporsi prediksi yang benar (baik positif, maupun negatif) dari seluruh data yang diuji. *Accuracy* tidak cocok digunakan untuk dataset yang tidak seimbang.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Precision bertujuan untuk mengukur seberapa banyak prediksi positif merupakan sebuah prediksi yang benar-benar positif dan bukan hanya *false positif.*

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Recall bertujuan untuk mengukur seberapa banyak dari prediksi positif sebenarnya berhasil dikenali oleh model. *Recall* penting digunakan untuk menghindari *false negatif*.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

F1-Score adalah rata-rata harmonik antara *Precision* dan *Recall*, metode ini bertujuan untuk menyeimbangkan keduanya, terutama untuk mengukur seberapa baik model mempertahankan keseimbangan antara precision dan *recall*.

ROC (*Receiver Operating Characteristic*) dan AUC (*Area Under the Curve*) merupakan metode evaluasi tambahan yang digunakan untuk menilai kemampuan model klasifikasi dalam membedakan antara kelas positif dan negatif. Kurva ROC mennggambarkan hubungan

antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) pada berbagai nilai ambang batas (*threshold*).

True Positive Rate (TPR) atau *Recall* menunjukkan seberapa besar proporsi data positif yang berhasil dikeali oleh model, dengan rumus:

$$Sumbu\ x(FPR) = \frac{FP}{FP+TN} \quad (5)$$

False Positive Rate (FPR) menunjukkan proporsi data negatif yang salah diklasifikasikan sebagai positif, dengan rumus:

$$Sumbu\ xY(TPR) = \frac{FP}{FP+TN} \quad (6)$$

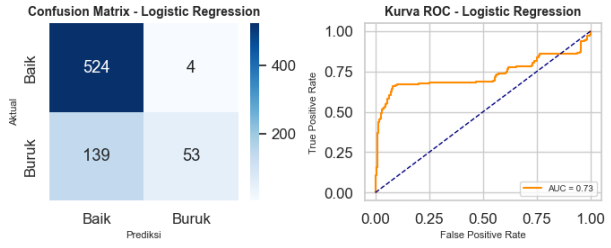
Kurva ROC dihasilkan dengan membuat plot nilai TPR terhadap FPR pada berbagai *threshold*, sehingga dapat memperlihatkan *trade-off* antara sensitivitas dan kesalahan prediksi positif. Semakin mendekati sudut kiri atas grafik, semakin baik kemampuan model dalam membedakan antara dua kelas.

Sementara itu, AUC (*Area Under the Curve*) mengukur luas area di bawah kurva ROC. Nilai AUC berkisar antara 0 hingga 1, di mana semakin mendekati 1 menunjukkan bahwa model memiliki kemampuan klasifikasi yang semakin baik. Secara umum, nilai AUC di atas 0.9 menunjukkan performa model yang sangat baik, antara 0.7-0.9 dianggap cukup baik. Sedangkan di bawah 0.7 menandakan model perlu di perbaiki.

3. Hasil dan Pembahasan

3.1 Hasil dari Metode *Logistic Regression*

Visualisasi hasil klasifikasi menggunakan metode *Logistic Regression* ditunjukkan pada Gambar 1, yang mencakup *confusion matrix* serta kurva ROC untuk menggambarkan performa model.



Gambar 1. Confusion Matrix dan Kurva ROC dari Hasil Klasifikasi Menggunakan Metode *Logistic Regression*

Dari grafik didapatkan 524 data "baik" dengan 4 false positif, dan 139 data "buruk" dengan 53 false negatif, dari kurva ROC dapat disimpulkan bahwa metode *logistic regression* mampu membedakan antara baik dan buruk dengan cukup baik.

Sementara itu, pada kurva ROC (*Receiver Operating Characteristic*), diperoleh nilai AUC (*Area Under Curve*) sebesar 0.71. Nilai ini menunjukkan bahwa model memiliki kemampuan klasifikasi yang cukup baik dalam membedakan antara kelas Baik dan Buruk. Semakin tinggi nilai AUC, semakin baik pula kemampuan model dalam membedakan dua kelas tersebut.

Selain itu, Tabel 1 menampilkan nilai *True Positive* Rate (TPR) dan *False Positive Rate* (FPR) pada berbagai nilai ambang batas (*threshold*) yang digunakan dalam pembentukan kurva ROC. Berdasarkan tabel tersebut, ketika *threshold* menurun dari 0.83 menjadi 0.63, nilai TPR meningkat dari 0.00 menjadi 0.15, yang menandakan bahwa semakin banyak data kelas *Buruk* yang berhasil dikenali oleh model. Namun, peningkatan TPR ini juga diikuti oleh sedikit kenaikan FPR dari 0.00 menjadi 0.007, artinya terdapat sebagian kecil data "baik" yang salah diklasifikasikan sebagai "buruk".

Tabel 1. Nilai TPR & FPR Metode Klasifikasi Logistic Regression

FPR	TPR	Threshold
0.000000	0.000000	inf
0.000000	0.005208	0.832643
0.000000	0.031250	0.828756
0.001894	0.031250	0.828583
0.001894	0.109375	0.822657
0.003788	0.109375	0.822314
0.003788	0.145833	0.637457
0.005682	0.145833	0.636827
0.005682	0.156250	0.636022
0.007576	0.156250	0.636009

Kondisi ini menggambarkan adanya *trade-off* antara sensitivitas (TPR) dan tingkat kesalahan klasifikasi (FPR). Dengan kata lain, semakin rendah nilai *threshold*, semakin sensitif model dalam mengenali kelas positif (*Buruk*), namun dengan risiko meningkatnya kesalahan dalam mengklasifikasikan data negatif (*Baik*). Pola ini menjadi dasar bentuk kurva ROC pada model *Logistic Regression*.

Berdasarkan hasil pengujian metode *Logistic Regression* yang disajikan pada Tabel 2, diperoleh nilai akurasi sebesar 0.801389, *precision* sebesar 0.929825, *recall* sebesar 0.276042, dan F1-score sebesar 0.425703. Nilai akurasi yang cukup tinggi menunjukkan bahwa model mampu melakukan klasifikasi dengan tingkat ketepatan yang baik secara keseluruhan. Namun, nilai *recall* yang rendah mengindikasikan bahwa model masih kurang optimal dalam mengenali seluruh data pada kelas positif.

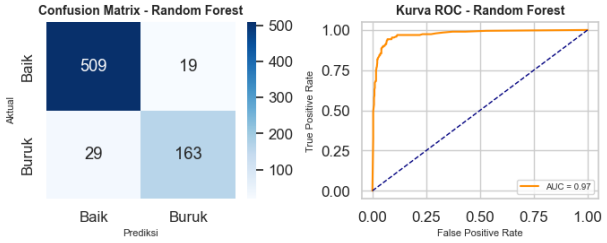
Meskipun demikian, nilai *precision* yang tinggi menunjukkan bahwa sebagian besar prediksi positif yang dihasilkan model tergolong benar. Secara keseluruhan, model mengklasifikasikan 663 data ke dalam kategori "baik" dan 57 data ke dalam kategori "buruk", sehingga dapat disimpulkan bahwa model memiliki kemampuan klasifikasi yang cukup baik namun perlu peningkatan pada aspek sensitivitas terhadap kelas positif.

Tabel 2. Hasil Evaluasi dari Klasifikasi Metode *Logistic Regression*

Accuracy	Precision	Recall	F1	Baik	Buruk
0.801389	0.929825	0.276042	0.425703	663	57

3.2 Hasil dari Metode *Random Forest*

Visualisasi hasil klasifikasi menggunakan metode *Random Forest* ditunjukkan pada Gambar 2, yang mencakup *confusion matrix* serta kurva ROC untuk menggambarkan performa model.



Gambar 2. Confusion Matrix dan Kurva ROC dari Hasil Klasifikasi Menggunakan Metode *Random Forest*

Dari grafik didapatkan 509 data "baik" dengan 19 false positif, dan 29 data "buruk" dengan 163 false negatif, dari kurva ROC dapat disimpulkan bahwa metode *random forest* mampu membedakan antara baik dan buruk lebih baik dibandingkan dengan metode *logistic regression.*

Pada kurva ROC (*Receiver Operating Characteristic*), diperoleh nilai AUC (*Area Under Curve*) sebesar 0.97, yang menunjukkan performa klasifikasi yang sangat baik. Nilai AUC yang mendekati 1.0 mengindikasikan bahwa model *Random Forest* mampu membedakan kelas "baik" dan "buruk" dengan tingkat akurasi yang tinggi. Dengan demikian, dapat disimpulkan bahwa *Random Forest* merupakan model yang paling optimal dan andal dalam mengklasifikasikan data pada penelitian ini.

Selain itu, Tabel 3, menampilkan nilai True Positive Rate (TPR) dan False Positive Rate (FPR) pada berbagai nilai ambang batas (*threshold*) yang digunakan dalam perhitungan kurva ROC. Berdasarkan tabel tersebut, terlihat bahwa ketika nilai *threshold* diturunkan dari 1.00 ke 0.90, nilai TPR meningkat secara bertahap dari 0.32 menjadi 0.61, menandakan semakin banyak data yang diklasifikasikan sebagai *Buruk* secara benar. Namun, hal ini juga disertai dengan sedikit peningkatan FPR dari 0.0037 menjadi 0.0094, yang berarti model sedikit lebih sering salah mengklasifikasikan data *Baik* sebagai *Buruk*. Pola kenaikan TPR dan FPR ini menunjukkan trade-off antara sensitivitas dan spesifisitas model yang divisualisasikan pada kurva ROC.

Tabel 3. Nilai TPR dan FPR Metode Klasifikasi Random Forest

FPR	TPR	Threshold
0.000000	0.000000	Inf
0.003788	0.328125	1
0.003788	0.395833	0.99
0.003788	0.432292	0.98
0.003788	0.458333	0.97
0.003788	0.494792	0.96
0.007576	0.536458	0.94
0.007576	0.578125	0.92
0.009470	0.593750	0.91
0.009470	0.609375	0.90

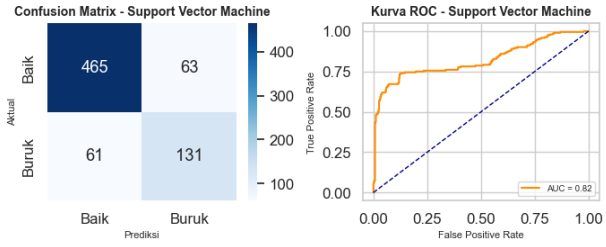
Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4, metode *Random Forest* menunjukkan performa yang sangat baik dalam melakukan klasifikasi. Model ini memperoleh akurasi sebesar 0.9333, *precision* 0.8956, *recall* 0.8489, dan F1-score sebesar 0.8716. Selain itu, jumlah prediksi benar untuk kelas "baik" mencapai 538, sedangkan untuk kelas "buruk" sebanyak 182. Nilai-nilai tersebut menunjukkan bahwa algoritma *Random Forest* mampu mengenali pola data dengan akurasi tinggi serta memiliki keseimbangan yang baik antara *precision* dan *recall*, sehingga menghasilkan performa klasifikasi yang stabil.

Tabel 4. Hasil Evaluasi dari Metode Klasifikasi *Random Forest*

Accuracy	Precision	Recall	F1	Baik	Buruk
0.933333	0.895604	0.848958	0.871658	538	182

3.3 Hasil dari Metode SVC

Visualisasi hasil klasifikasi menggunakan metode SVC ditunjukkan pada Gambar 3, yang mencakup *confusion matrix* serta kurva ROC untuk menggambarkan performa model.



Gambar 3. Confusion Matrix dan Kurva ROC dari Hasil Klasifikasi Menggunakan Metode *SVC*

Dari grafik didapatkan 465 data "baik" dengan 63 false positif, dan 61 data "buruk" dengan 131 *false*

negatif, dari kurva ROC dapat disimpulkan bahwa metode *logistic regression* mampu membedakan antara baik dan buruk dengan cukup baik.

Pada kurva ROC (*Receiver Operating Characteristic*) diperoleh nilai AUC (*Area Under Curve*) sebesar 0.82, yang menunjukkan performa klasifikasi yang baik. Nilai AUC yang mendekati 1.0 mengindikasikan bahwa model SVC memiliki kemampuan yang cukup tinggi dalam membedakan antara kelas positif (baik) dan negatif (buruk). Dengan demikian, dapat disimpulkan bahwa metode SVC menunjukkan performa yang solid dan andal dalam mengklasifikasikan data pada penelitian ini, meskipun masih terdapat beberapa kesalahan klasifikasi.

Selain itu, Tabel 5 menampilkan nilai True Positive Rate (TPR) dan False Positive Rate (FPR) pada berbagai nilai ambang batas (*threshold*) yang digunakan dalam pembentukan kurva ROC. Berdasarkan tabel tersebut, terlihat bahwa ketika *threshold* menurun dari 0.98 ke 0.69, nilai TPR meningkat dari 0.005 menjadi 0.43, yang menunjukkan peningkatan kemampuan model dalam mengenali kelas *Buruk* secara benar. Namun, hal ini juga menyebabkan kenaikan FPR dari 0.00 menjadi 0.009, menandakan bahwa semakin banyak data *Baik* yang salah terklasifikasi sebagai *Buruk*. Pola ini menunjukkan adanya trade-off antara sensitivitas dan spesifisitas, yang menjadi dasar bentuk kurva ROC pada model SVM.

Tabel 5. Nilai TPR dan FPR Metode Klasifikasi Support Vector Machine (SVC)

FPR	TPR	Threshold
0.000000	0.000000	Inf
0.000000	0.005208	0.982196
0.000000	0.057292	0.972272
0.001894	0.057292	0.971930
0.001894	0.062500	0.958311
0.009788	0.062500	0.954215
0.003788	0.072917	0.939174
0.007576	0.072917	0.929934
0.007576	0.432292	0.695119
0.009470	0.432292	0.694096

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 6, metode *Support Vector Classifier (SVC)* menghasilkan performa yang cukup baik namun masih berada di bawah hasil yang dicapai oleh *Random Forest*. Model SVC memperoleh akurasi sebesar 0.8278, *precision* 0.6753, *recall* 0.6823, dan F1-score sebesar 0.6788. Adapun jumlah prediksi benar untuk kelas "baik*"* adalah 526 dan untuk kelas "buruk" sebanyak 194. Hasil ini menunjukkan bahwa meskipun SVC mampu melakukan klasifikasi dengan tingkat ketepatan yang cukup baik, namun algoritma ini belum seoptimal *Random Forest* dalam mengenali pola data dan menjaga keseimbangan antara precision dan recall.

Tabel 6. Hasil Evaluasi dari Klasifikasi Metode SVC

Accuracy	Precision	Recall	F1	Baik	Buruk
0.827778	0.675258	0.682292	0.678756	526	194

3.4 Perbandingan *Logistic Regression*, *Random Forest*, dan SVC.

Berdasarkan hasil pengujian pengujian yang ditampilkan pada Tabel 7, dilakukan perbandingan performa antara tiga algoritma klasifikasi yaitu *Logistic Regression*, *Random Forest*, dan *Support Vector Classifier* (SVC). Dari hasil tersebut, dapat dilihat bahwa *Random Forest* menunjukkan performa terbaik dengan akurasi sebesar 0.9333, *precision* 0.8956, *recall* 0.8489, dan F1-score sebesar 0.8717. Nilai-nilai ini menunjukkan keseimbangan yang baik antara kemampuan model dalam mengenali kelas positif maupun negatif. Selain itu, nilai AUC sebesar 0.9734 mengindikasikan bahwa *Random Forest* memiliki kemampuan yang sangat baik dalam membedakan antara kelas "baik*"* dan "buruk", dengan tingkat akurasi yang hampir mendekati sempurna.

Sementara itu, *Support Vector Classifier* (SVC) menempati posisi kedua dengan akurasi 0.8278, *precision* 0.6753, *recall* 0.6823, F1-score 0.6788, dan AUC 0.8207. Nilai AUC yang cukup tinggi menunjukkan bahwa SVC memiliki kemampuan diskriminatif yang baik, meskipun performanya masih berada di bawah *Random Forest* karena tingkat *precision* dan *recall* yang relatif lebih rendah.

Adapun *Logistic Regression* memperoleh hasil yang paling rendah dengan akurasi 0.8014, *precision* 0.9298, *recall* 0.2760, F1-score 0.4257, dan AUC 0.7305. Nilai recall yang rendah menunjukkan bahwa model ini kurang mampu mendeteksi seluruh data positif dengan baik, meskipun *precision*-nya tinggi. Nilai AUC yang lebih kecil dibandingkan dua model lainnya juga menandakan bahwa *Logistic Regression* memiliki kemampuan pemisahan kelas yang lebih lemah.

Secara keseluruhan, hasil ini menunjukkan bahwa *Random Forest* merupakan algoritma yang optimal dalam melakukan klasifikasi pada penelitian ini karena memberikan performa paling seimbang dan nilai AUC tertinggi di antara ketiga model yang dibandingkan.

Tabel 7. Perbandingan Hasil Evaluasi *Logistic Regression*, *Random Forest*, dan SVC.

Model	Accuracy	Precision	Recall	F1	AUC
Logistic Regression	0.801389	0.929825	0.276042	0.425703	0.730518

Random Forest	0.933333	0.895604	0.848958	0.871658	0.973411
SVR	0.827778	0.675258	0.682292	0.678756	0.820722

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan tiga algoritma machine learning yaitu Random Forest, Support Vector Classifier (SVC), dan Logistic Regression mampu memberikan hasil klasifikasi yang cukup baik terhadap kualitas air sungai di Jakarta. Di antara ketiganya, algoritma Random Forest menunjukkan kinerja paling unggul dengan tingkat akurasi mencapai 93,33% dan nilai F1-score sebesar 0,87, yang mencerminkan keseimbangan terbaik antara presisi dan recall. Sementara itu, Logistic Regression memiliki nilai presisi tertinggi namun belum optimal dalam mengenali seluruh data positif, sedangkan SVC menunjukkan performa yang relatif stabil dengan tingkat akurasi menengah di antara kedua model lainnya.

Kelebihan utama dari Random Forest terletak pada kemampuannya menangani data non-linear dan menghasilkan model yang adaptif terhadap variasi data. Di sisi lain, Logistic Regression unggul dalam efisiensi komputasi serta penggunaan sumber daya, namun kurang sesuai untuk data dengan hubungan yang kompleks. Adapun SVC memiliki kemampuan yang baik dalam memisahkan kelas, tetapi membutuhkan waktu pelatihan yang lebih lama. Berdasarkan hasil tersebut, dapat disimpulkan bahwa Random Forest merupakan algoritma yang paling tepat digunakan dalam klasifikasi kualitas air sungai, karena mampu memberikan keseimbangan terbaik antara akurasi dan efisiensi komputasi.

Sebagai arah pengembangan selanjutnya, disarankan untuk memperluas cakupan dataset dengan menambahkan data dari wilayah sungai lainnya serta parameter lingkungan tambahan seperti kadar logam berat dan nitrat. Selain itu, penerapan teknik hyperparameter tuning dan feature selection dapat membantu meningkatkan performa model dengan menyoroti parameter yang paling berpengaruh terhadap hasil klasifikasi. Penggunaan metode cross-validation yang lebih menyeluruh juga dapat meminimalkan risiko overfitting. Ke depan, penelitian ini berpotensi dikembangkan dengan pendekatan deep learning atau integrasi data spasial dan temporal guna menghasilkan model prediksi yang lebih akurat dan aplikatif terhadap kondisi lingkungan yang dinamis.

REFERENSI

[1] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.)*. O'Reilly Media, 2023.

[2] A. W. Saputra, F. H. Arifianto, and R. H. Putra, "Application of Machine Learning Models in Water Quality Classification in Lake Maninjau: Random Forest as the Optimal Solution," *E-Tech J. Ilm. Teknol. Elektro*, vol. 10, no. 2, pp. 45–54, 2023.

[3] A. P. Singh and M. N. Islam, "Optimization of Machine Learning Algorithms for River Water Quality Prediction," *Environ. Monit. Assess.*, vol. 196, no. 3, pp. 1–15, 2024.

[4] J. K. Nair and A. Thomas, "Comparative Evaluation of Ensemble Learning Techniques for Surface Water Quality Classification," *Water Resour. Manag.*, vol. 37, no. 4, pp. 1103–1118, 2023.

[5] S. Li and X. Zheng, "Cryptanalysis of a Chaotic Image Encryption Method," in *Proceedings IEEE – ISACS*, Scottsdale, Arizona, 2002.

[6] Y. Mao and G. Chen, "Chaos-Based Image Encryption BT - Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neural Computing and Robotics," Berlin: Springer, 2003.

[7] S. Khan, R. Kumar, and P. Mehta, "Water Quality Assessment Through Predictive Modeling Employing Machine Learning Methods," *J. Comput. Biomed. Informatics*, vol. 3, no. 1, pp. 33–42, 2024.

[8] M. A. Yildiz and B. Kaya, "Prediction of Water Quality's pH Value Using Random Forest and LightGBM Algorithms," *MEMBA J. Water Sci.*, vol. 6, no. 2, pp. 120–130, 2025.

[9] T. Y. Nguyen, H. Pham, and N. Tran, "Deep Learning-Based Framework for Water Quality Forecasting in Urban Rivers," *J. Environ. Manage.*, vol. 339, p. 118112, 2023.

[10] F. M. Rahman and M. Chowdhury, "Comparative Study of Machine Learning Techniques for Predicting Water Quality in Developing Regions," *Heliyon*, vol. 9, no. 1, p. e13211, 2023.

[11] A. Elbeltagi, M. E. Salah, and H. E. Abdelrahman, "Random Forest and Logistic Regression Algorithms for Prediction of Groundwater Contamination Using Ammonia Concentration," *Arab. J. Geosci.*, vol. 15, no. 12, pp. 1–15, 2022.

[12] N. Hassan, M. Salleh, and A. Rahman, "Predicting Kereh River's Water Quality: A Comparative Study of Machine Learning Models," *Environ. Proc. J.*, vol. 7, no. 21, pp. 49–56, 2022.

[13] L. Chen, X. Zhao, and Y. Fang, "Improved Random Forest Model for Water Quality Index Prediction," *Environ. Sci. Pollut. Res.*, vol. 31, no. 1, pp. 87–99, 2024.

[14] D. P. Patel, A. Mishra, and R. Shah, "Performance Analysis of Supervised Machine Learning Algorithms for River Water Quality Prediction," *Sustainability*, vol. 14, no. 6, pp. 1–12, 2022.

[15] H. Zhang and J. Li, "Predicting River Water Quality Using Hybrid Random Forest and Gradient Boosting Models," *Environ. Model. Softw.*, vol. 172, p. 106233, 2025.

[16] D. Sharma and R. Joshi, "Evaluation of Water Quality Classification Using Random Forest and SVM: A Case Study of Ganga River Basin," *Sci. Rep.*, vol. 12, no. 2113, pp. 1–10, 2024.

[17] K. R. Castleman, *Digital Image Processing*. New Jersey: Prentice Hall, 1998.

[18] B. Wiv, C. Liv, and B. Jonathan, "Statistics Review 14: Logistic Regression," *Natl. Libr. Med.*, 2005.

[19] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[20] C. Jair, F. García-Lamont, L. Rodríguez-Mazahua, and A. López, "A Comprehensive Survey on Support Vector Machine Classification: Applications, Challenges and Trends," *Appl. Comput. Informatics*, vol. 17, no. 1, pp. 1–22, 2020.